IN-36
145549
P.17

NASA Technical Memorandum 105994

A Fuzzy Logic Based Controller for the Automated Alignment of a Laser-Beam-Smoothing Spatial Filter

M.J. Krasowski
Lewis Research Center
Cleveland, Ohio

and

D.E. Dickens
University of Dayton
Dayton, Ohio

December 1992

(NASA-TM-105994) A FUZZY LOGIC BASED CONTROLLER FOR THE AUTOMATED ALIGNMENT OF A LASER-BEAM-SMOOTHING SPATIAL FILTER (NASA) 17 p

N93-18091

Unclas

G3/36 0145549

NASA

A FUZZY LOGIC BASED CONTROLLER FOR THE AUTOMATED ALIGNMENT OF A LASER-BEAM-SMOOTHING SPATIAL FILTER

M.J. Krasowski
NASA-Lewis Research Center
21000 Brookpark Rd.
Cleveland, Ohio 44135

D.E. Dickens
Electrical Engineering Department
University of Dayton
Dayton, Ohio 45469

ABSTRACT

A fuzzy logic based controller for a laser-beam-smoothing spatial filter is described. It is demonstrated that a human operator's alignment actions can easily be described by a system of fuzzy rules of inference. The final configuration uses inexpensive, off-the-shelf hardware and allows for a compact, readily implemented embedded control system.

I. INTRODUCTION

Optical measurement systems using lasers require a number of optical components which must be aligned by a skilled technician. Certain applications for these systems place them in environments which may be inaccessible to or too harsh for human operators. In these situations, it is desirable that controllers be developed to automate the alignment process. We describe a fuzzy logic based controller which effects the alignment of a laser-beam-smoothing spatial filter.

In a typical alignment problem, a skilled operator views an optical pattern, interprets it and then performs an alignment action by adjusting the appropriate axis of a component. Typically a series of adjustments over a number of degrees of freedom are required before the person is satisfied with the results. An automated alignment system that mimics a human's control actions must possess vision, the ability to intepret the imaged scene and also the means to physically adjust the components involved in the problem. In our system, a CCD video camera, interfaced to a Fuzzy Pattern Comparator Applications System (1), provides vision and simple feature extraction. A Fuzzy Microcontroller Development System (2) provides the interpretive capabilities and the choice of an appropriate set of control outputs. A custom built, three channel, bidirectional, variable speed motor controller uses the fuzzy controller outputs to drive the actuators for the x, y and z axes of the spatial filter. A block diagram of our system is given in figure 1.

Earlier work studied the use of artificial neural networks to learn the sequence of steps an operator uses to achieve an alignment, (3) and the results were very promising. The decision to use fuzzy logic in this work instead of neural nets was based on a number of considerations. First of all, neural net hardware and development tools are relatively expensive and complex. Neural net chips may cost several hundred dollars whereas the fuzzy microcontroller used in this design costs about ten dollars. Also, a great deal of pre-and post-processing is normally required to normalize and unnormalize input and output data for neural nets. This may require extra processor and software overhead. The fuzzy controller can take data in any form which fits into the word length of the device since input mappings are achieved in the fuzzification process. Outputs are exclusively associated with the fuzzy controller's winning rule and so can take on any useful form the designer wishes. Finally, we felt that since we had to go through the process of observing a skilled operator's control actions in order to create a training set for a neural net, we could also typify those actions with a set of fuzzy rules.

This paper will not serve as a primer on fuzzy logic. The reader is directed to references 4,5 and 6 which cover various fuzzy control techniques very well. Though we describe the type of fuzzy inference used by the fuzzy microcontroller and pattern comparator used in this work, it is advised that the reader secure the appropriate data sheets from the manufacturer (references 7 and 8).

Section II we give an overview of the system. Section III covers the vision system and how it provides information on beam position and area of illumination. Section IV describes the motor controller which drives the x,y and z actuators based on the fuzzy microcontroller's control variables. Section V briefly describes the fuzzy microcontroller and how it interprets the outputs of the vision system to determine the control variables. Section VI lists our results.

II. SYSTEM OVERVIEW

A spatial filter, Fig. 2, is used here to filter out light which is not propagating within the laser's primary spatial mode. It consists of a lens and a pinhole. The pinhole is mounted in the filter's xy translational stage and the lens in the z stage. A laser beam is directed at the lens and the position of the xy stage (centering) and the z stage (focusing) are adjusted until the light exiting the pinhole and falling on a translucent target has reached maximum intensity and there is no evidence of diffraction rings. At this point, the laser beam is in focus at the center of the pinhole and the filter is aligned.

In this example, the fuzzy control algorithm concerns itself with rules based on a human operator's actions and the resultant outcomes. Watching an operator align the filter, it became apparent that to center the beam, one needed to move the pinhole across x and y in the same direction that the beam is to move. So, we came up with our first set of general rules: if the beam is off to the left, move it right; if the beam is off to the right move it left; if it is high, move it down; and if it is low, move it up.

Relative to focusing, if the beam appears to be nearly centered, move z in towards the pinhole to expand the exiting beam pattern. This becomes our next general rule.

When the beam is centered and the light falling on the target uniformly covers a maximum area, then quit. This is the form of our last general rule.

For the purpose of this work, an Intel based PC is used as the system platform. In this manner, the various components of the control system are glued together via the system bus and applications software written in Pascal. If this system is ever targeted to an embedded application, the functions of the PC can be relegated to any of a number of general purpose microprocessors or microcontrollers. A microcontroller can also provide the pulse-width-modulated (PWM) signals for the motor drivers.

III. VISION SYSTEM

The purpose of the vision system is to provide the fuzzy microcontroller with measurements on the laser spot's size and position. The laser spot illuminates a translucent plate. A CCD video camera positioned behind the plate in line with the beam images the laser spot which is then digitized by a frame grabber. The Fuzzy Pattern Comparator Applications System has on board, together with the high speed comparator, a frame grabber with user selectable resolution of one to six bits and an effective 128 by 116 pixel array. This system allows for the incoming video image to be compared with up to eight stored reference patterns. These comparisons are done in parallel to provide video speed measurements of the Euclidean distance (or Hamming distance for the one bit case) calculated as the accumulated differences between the incoming and stored pixels.

In the comparator's standard configuration, if the input pattern's distance to the closest reference pattern is less than a user specified threshold (that is, its fuzzy strength of membership suggests that the incoming image is a member of the set of images typified by that pattern) then that reference pattern is labeled the winner. We decided to use the comparator in a slightly different manner than this. The incoming image is compared with five reference patterns and the calculated strengths with which the image is associated with these references are used to infer spot size and position.

As a video image is pulled in from the camera, it is digitized with a one bit resolution. We found that even with six bits we could not adequately image any of the diffraction rings as they were quite dim relative to the center spot. Being that the target shape of the filtered beam is gaussian, a measurement which associates intensity with cross-sectional area is adequate for our purposes. A one bit conversion produces the size of the beam at the point where the intensity exceeds a specific threshold. This threshold level is a function of the iris setting of the camera and of the reference voltage at the A/D converter. The voltage reference is factory set and we did not concern ourselves with it. The iris was adjusted to minimize blooming in the camera and to maximize spot size for a digitized image of an aligned spot and then locked in place.

As the image is digitized, it is compared, pixel for pixel, with each of the five reference patterns. Each time that there is a difference in pixel state between the incoming image and the stored pattern, an error accumulator associated with that pattern is incremented. Upon completion of the pattern comparison, the accumulator value represents the Hamming distance between the stored pattern and the incoming image.

The size of the illuminated spot is determined by comparing the incoming image I, with reference pattern Ps, which is loaded with a "0" in all 14,848 pixels. The incoming image I's strength of membership with Ps, $mu_{Ps}(I)$, is defined to be equal to Hs, which is the Hamming distance between the two images and is equal to the number of pixels in the incoming image which have a value of "1".

Since the fuzzy microcontroller requires eight bit inputs, $mu_{Ps}(I)$ is scaled to obtain a measured spot size, w, that is normalized relative to 127 (half scale for an eight bit word),

$$w = 127 * (mu_{Ps}(I)/mu_{Ps0}(I)),$$

where $mu_{Ps0}(I)$ is an experimentally determined target value for $mu_{Ps}(I)$, i.e. the $mu_{Ps}(I)$ that was measured after the spatial filter was manually aligned.

The remaining four patterns, Pa, Pb, Pc and Pd, provide information on the location of the laser spot's centroid by acting as fuzzy quadrant detectors. The four patterns represent quadrants a, b, c and d, respectively, which are arbitrarily positioned as follows:

```
b | a
-----
d | c .
```

Each pattern is loaded with a "0" for each pixel in the quadrant, and a "1" everywhere else. In this manner the Hamming distance between the incoming image and Pq, where q is either a, b, c or d is given by

$$Hq = Ho + Nin - Nout$$

where Ho is the number of pixels outside the quadrant, or 11136, Nin is the number of illuminated pixels inside the quadrant and Nout is the number of illuminated pixels outside the quadrant. Noting that

$$Hs = Nin + Nout,$$

the strength of membership a spot of size Hs has to a given quadrant q is given by

$$mu_{Pq}(I) = 50 - (50 * (Ho - Hq))/Hs,$$

where $mu_{Pq}(I)$ is an integer in the range 0 to 100.

These four $mu_{Pq}(I)$ values are used to locate the spot's centroid on the Pq image plane for input to the fuzzy microcontroller as x, y coordinates centered at 127, 127 as follows

$$x = 127 + (mu_{Pa}(I) + mu_{Pc}(I))/2 - (mu_{Pb}(I) + mu_{Pd}(I))/2$$

and

$$y = 127 + (mu_{Pa}(I) + mu_{Pb}(I))/2 - (mu_{Pc}(I) + mu_{Pd}(I))/2.$$

IV. MOTOR CONTROLLER

The motors actuating the three axes of the spatial filter are powered by a three channel, bidirectional, variable speed motor controller.

The fuzzy microcontroller, which will be described in section V, outputs a control variable eight bits in length. The lower nibble, bits 0 to 3, contains a motor speed with sixteen possible levels from 0, which means "stop", to 15, which means "full speed". The next two bits define motor direction, bit 5 set for forward, bit 4 set for reverse and bit 4 = bit 5 for stop. Bits 6 and 7 are unused.

The fuzzy microcontroller picks a speed and direction for each axis and these values are passed on to the appropriate motor channel via the PC.

The controller board shown in figure 3 is mapped into the PC I/O address space starting at 384 hex with the x axis port at 384, the y axis at 385 and z at 386. This address block puts the motor controller out of conflict with the two fuzzy development system boards. A schematic is included as figure 4.

Each channel is made up of a motor speed and direction register followed by a PWM stage and steering logic.

The PWM stage compares the speed nibble with a free running four bit counter to create the motor drive signal. For counter values less than the speed nibble value, power is delivered to the motor. For all other counter values the motor is off. For example, if the speed nibble is a 5 then the motor will be on for the first 5 counts and off for the next 11, yielding a pulse width of 5/16. This comparison and pulse generation is achieved by the 7485 magnitude comparators.

The steering logic utilizes the direction bits to steer the drive pulses to the proper half of the H-bridge motor drive made up of the 2N3906 transistors and the ULN2003. If the motor is to turn forward, current is steered through it in one direction. For reverse, current is steered in the other.

The x and y axes are driven by motorized linear actuators, and the z axis is driven by a dc motor coupled to it via a timing belt.

V. FUZZY MICRO CONTROLLER

The fuzzy microcontroller accepts the fuzzy comparator's outputs, w, x and y, and processes these values to obtain Cz, Cx and Cy, the eight bit variables which serve as the inputs to the motor drivers. For each of these three outputs, the fuzzy microcontroller contains a set of rules. The microcontroller inputs are first fuzzified using membership functions, which have user defined center-locations and half-widths. The degrees of association between the controller inputs and the membership functions are then used to determine the winning rule out of each set. The winning rule then supplies the value of the associated controller output.

The membership functions fuzzify each of the controller's inputs by first measuring the distance between the input and the function's center location. The input's degree of membership in that function is then given by 31 minus its distance to the function's center, unless the distance exceeds 30 or the specified half-width, in which case the degree of membership is zero. The degree of membership, then, ranges between 31, when the input is at the function's center location, to zero, which indicates no association between the input and that function.

This device allows up to 63 rules, which are grouped according to the output that each rule would control if it were selected as the winner. Each rule contains one or more fuzzified inputs, i.e. it specifies controller inputs together with the associated membership functions. Every rule also contains the crisp value to be taken by the associated controller output if that rule is determined to be the winner.

Each rule is evaluated by determining which of its prescribed fuzzy inputs yields the minimum degree of association. Then, out of each of the three rule sets, the rule that has the largest of these values is declared a winner. The crisp value contained within each winning rule is then written to the associated microcontroller output.

Based on the observations and general rules developed earlier, we create membership functions and fuzzy rules.

The choice of membership functions is somewhat arbitrary since they represent linguistic interpretations of the problem made by a human expert. Different people may solve or describe the same problem differently. For instance, we chose to fuzzify x and y axis position variables by the membership functions

BIGNEGATIVE, SMALLNEGATIVE, ZERO, SMALLPOSITIVE and BIGPOSITIVE. Our concept behind centering is to move the beam until x and y position are members of the fuzzy sets ZERO. So, for example, if x axis position is BIGPOSITIVE then move x fast negative. Rules of this sort are created for x and y.

Size is transformed by the membership functions GONE, SMALL, MEDIUM, LARGE, FOCUSED and TOOBIG. Rules are developed in a like manner to those for x and y.

The rules are based on the set of general alignment rules developed earlier. Rules are added and membership functions adjusted until satisfactory results are obtained. There are 49 rules in all.

Membership functions and rules are listed in table 1.

VI. RESULTS

If given an illuminated pixel count which represents a valid aligned state for the in-place optical system and the spatial filter under control, the system will always reach that state after a misalignment in the filter has occured. Misalignment consists of arbitrarily moving x and y to offset the beam and of backing out the z axis away from the pinhole. No misalignment whereby the spot disappears to the CCD camera is allowed. Figure 5 shows some initial states of misalignment and the final aligned conditions as directed by the fuzzy controller. The results were evaluated by a skilled operator who judged them as "good" alignments.

Individual alignments typically take about 15 seconds with the motor speeds used. No attempts were made to minimize the number of rules or to optimize the shapes and locations of the membership functions. Nor was any attempt made to optimize the system for speed.

CONCLUSIONS AND FUTURE WORK

We have shown that a fuzzy logic controller can be designed based on a skilled operator's actions and built utilizing inexpensive and easy to use development systems.

The example of the laser-beam-smoothing spatial filter yielded excellent results. The filter achieved alignments on par with that of the technician. Unlike the technician, who can only adjust one axis at a time, the parallel nature of the fuzzy microcontroller allowed for all three axes to align simultaneously.

The PC can be replaced by a general purpose microprocessor or microcontroller. This will reduce the size and cost of the system making it attractive as an embedded control system for remote optical system.

The gaussian shape of the filtered beam allowed us to use a very simple, single bit conversion and feature extraction scheme which resulted in a compact, inexpensive system utilizing readily available hardware. More complex illumination patterns may require more of the capabilities of the fuzzy comparator. These devices can be cascaded to allow for up to 256 simultaneous comparisons with no degradation in speed. A system thus configured and using an 8 bit frame grabber, could compare 256 4096 x 4096, 256 level per pixel images at video speed. Other hardware or software feature extraction techniques, to include neural networks, can also be utilized alone or together with this type of system for increased capabilities. Of course, as the complexity increases, so may the size and cost.

Future work will include the alignment of other optical components. Configurations in which one CCD camera is used in the sequential alignment of more than one optical component in a beam path would have an economical and size advantage.

References:

1. Fuzzy Pattern Comparator Applications System vended by American Neurologix, Inc., 411 Central Park Dr. Sanford, Florida 32771
2. Fuzzy Microcontroller Development System vended by American Neurologix, Inc., 411 Central Park Dr. Sanford, Florida 32771
3. Decker, A.J. and Krasowski, M., "General Procedure for Using Artificial Neural Networks to Automate the Alignment of Optical Components in Harsh Environments", The Advanced Earth-to-Orbit Propulsion Technology Conference, May 19-21, 1992.
4. Sugeno, Michio, "An Introductory Survey of Fuzzy Control", Information Sciences, Vol. 36, 1985, pp. 59-83.
5. Brubaker, David, "Fuzzy-logic basics: intuitive rules replace complex math", EDN, June 18, 1992, Vol. 37, No. 13, pp. 111-116.
6. Brubaker, David and Sheerer, Cedric, "Fuzzy-logic system solves control problem", EDN, June 18, 1992, Vol. 37, No. 13, pp. 121-128.
7. NLX230 Fuzzy Microcontroller, vended by American Neurologix, Inc., 411 Central Park Dr. Sanford, Florida 32771
8. NLX110 Fuzzy Pattern Comparator, vended by American Neurologix, Inc., 411 Central Park Dr. Sanford, Florida 32771

Table 1

Input	Fuzzifier	Center	Half-Width
Xin	Bignegative	84	16
Xin	Smallnegative	108	16
Xin	Zero	127	4
Xin	Smallpositive	146	16
Xin	Bigpositive	170	16
Yin	Bignegative	84	16
Yin	Smallnegative	108	16
Yin	Zero	127	4
Yin	Smallpositive	146	16
Yin	Bigpositive	170	16
Size	Gone	0	0
Size	Small	11	28
Size	Medium	56	21
Size	Large	101	28
Size	Focused	132	5
Size	Toobig	159	22

RULES

Rules, for example, may take the form -

IF Yin IS Smallnegative AND Size IS Small THEN 19.

This means that if input Yin is associated with membership function Smallnegative and if input Size is associated with membership function Small and if it is the winning rule then the output associated with this rule shall have a 19 written to it, (upper nibble = 1, lower nibble = 3) causing the motor to run in reverse at a speed of 3/16 of maximum.

RULES FOR Z AXIS OUTPUT:

IF	Size	IS	Focused	THEN 0
IF	Size	IS	Large	THEN 39
IF	Size	IS	Medium	THEN 42
IF	Xin	IS	Bignegative	THEN 0
IF	Xin	IS	Bigpositive	THEN 0
IF	Yin	IS	Bignegative	THEN 0
IF	Yin	IS	Bigpositive	THEN 0
IF	Size	IS	Gone	THEN 30

```
IF  Size IS Toobig        THEN 30
IF  Xin  IS Smallnegative AND Size IS Small   THEN 44
IF  Xin  IS Zero          AND Size IS Small   THEN 44
IF  Xin  IS Smallpositive AND Size IS Small   THEN 44
IF  Yin  IS Smallnegative AND Size IS Small   THEN 44
IF  Yin  IS Zero          AND Size IS Small   THEN 44
IF  Yin  IS Smallpositive AND Size IS Small   THEN 44
```

RULES FOR X AXIS OUTPUT:

```
IF  Xin  IS Bignegative   AND Size IS Small   THEN 20
IF  Xin  IS Bignegative   AND Size IS Medium  THEN 19
IF  Xin  IS Bignegative   AND Size IS Large   THEN 18
IF  Xin  IS Bignegative   AND Size IS Focused THEN 18
IF  Xin  IS Smallnegative AND Size IS Small   THEN 19
IF  Xin  IS Smallnegative AND Size IS Medium  THEN 19
IF  Xin  IS Smallnegative AND Size IS Large   THEN 18
IF  Xin  IS Smallnegative AND Size IS Focused THEN 18
IF  Xin  IS Zero          THEN 0
IF  Xin  IS Smallpositive AND Size IS Small   THEN 35
IF  Xin  IS Smallpositive AND Size IS Medium  THEN 35
IF  Xin  IS Smallpositive AND Size IS Large   THEN 34
IF  Xin  IS Smallpositive AND Size IS Focused THEN 34
IF  Xin  IS Bigpositive   AND Size IS Small   THEN 36
IF  Xin  IS Bigpositive   AND Size IS Medium  THEN 35
IF  Xin  IS Bigpositive   AND Size IS Large   THEN 34
IF  Xin  IS Bigpositive   AND Size IS Focused THEN 34
```

RULES FOR Y AXIS OUTPUT:

```
IF  Yin  IS Bignegative   AND Size IS Small   THEN 20
IF  Yin  IS Bignegative   AND Size IS Medium  THEN 19
IF  Yin  IS Bignegative   AND Size IS Large   THEN 18
IF  Yin  IS Bignegative   AND Size IS Focused THEN 18
IF  Yin  IS Smallnegative AND Size IS Small   THEN 19
IF  Yin  IS Smallnegative AND Size IS Medium  THEN 19
IF  Yin  IS Smallnegative AND Size IS Large   THEN 18
IF  Yin  IS Smallnegative AND Size IS Focused THEN 18
IF  Yin  IS Zero          THEN 0
IF  Yin  IS Smallpositive AND Size IS Small   THEN 35
IF  Yin  IS Smallpositive AND Size IS Medium  THEN 35
IF  Yin  IS Smallpositive AND Size IS Large   THEN 34
IF  Yin  IS Smallpositive AND Size IS Focused THEN 34
IF  Yin  IS Bigpositive   AND Size IS Small   THEN 36
IF  Yin  IS Bigpositive   AND Size IS Medium  THEN 35
IF  Yin  IS Bigpositive   AND Size IS Large   THEN 34
IF  Yin  IS Bigpositive   AND Size IS Focused THEN 34
```

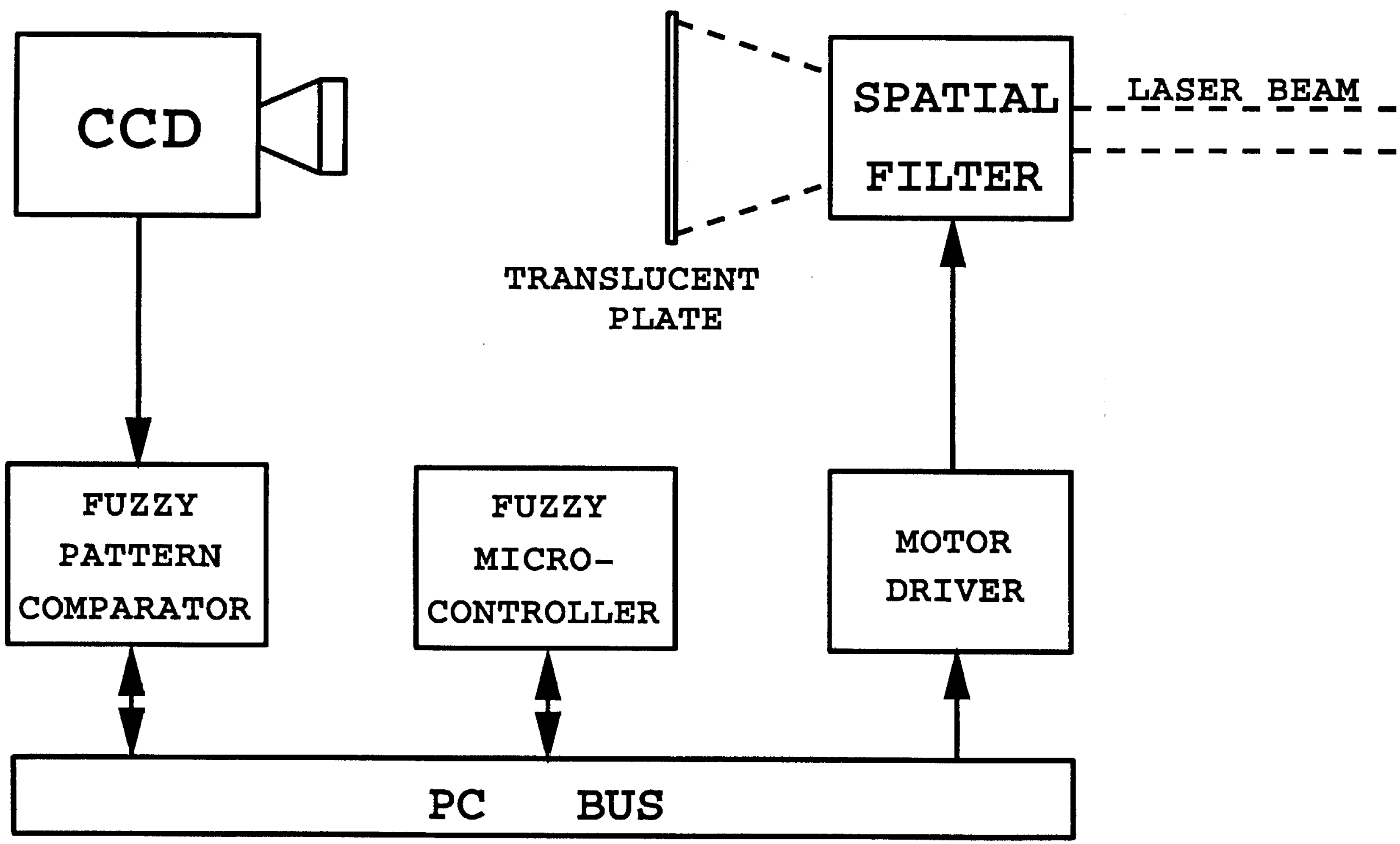


Figure 1.—Block diagram of system.

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

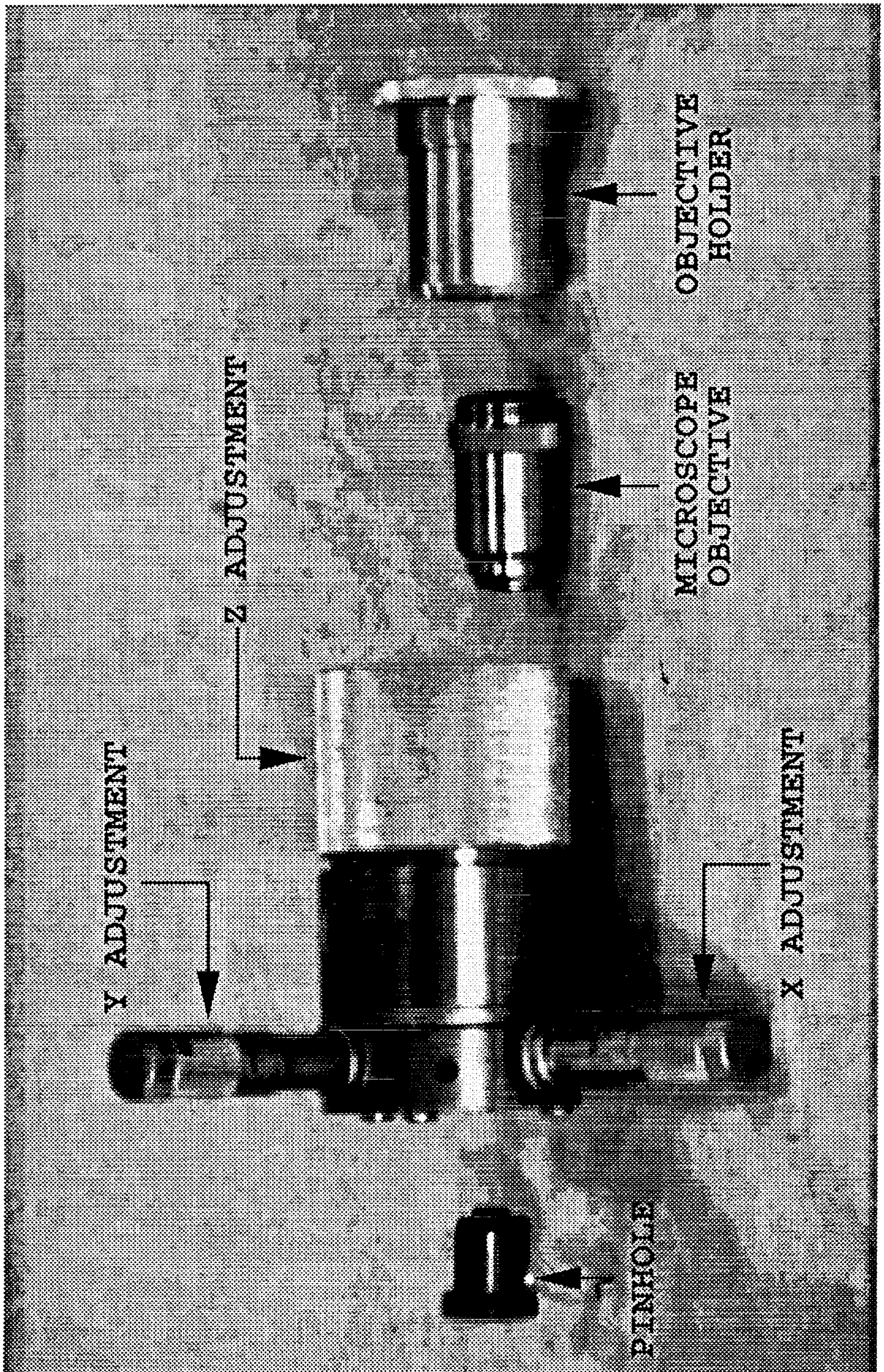


Figure 2.—A disassembled spatial filter assembly.

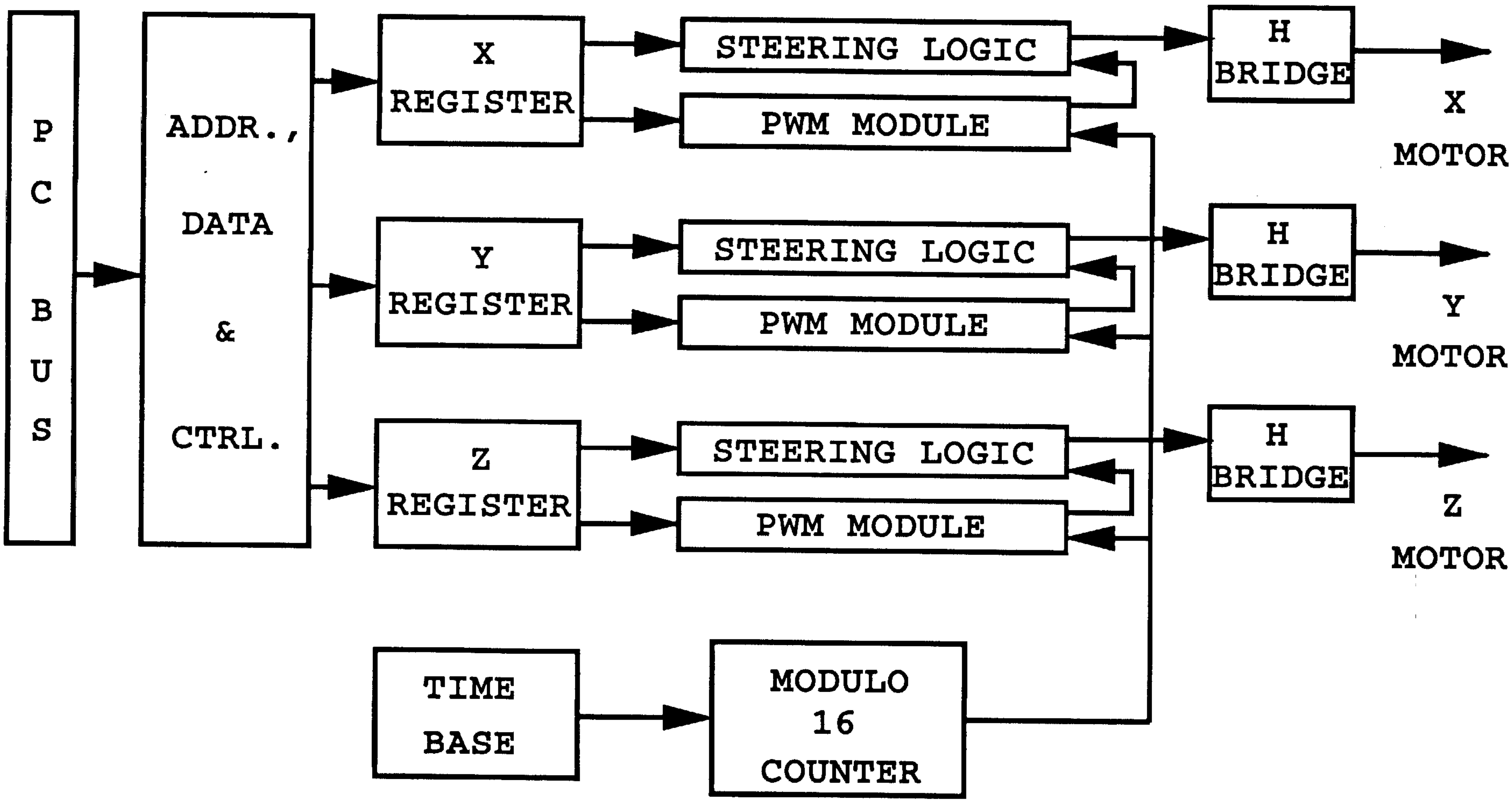


Figure 3.—Block diagram of motor driver.

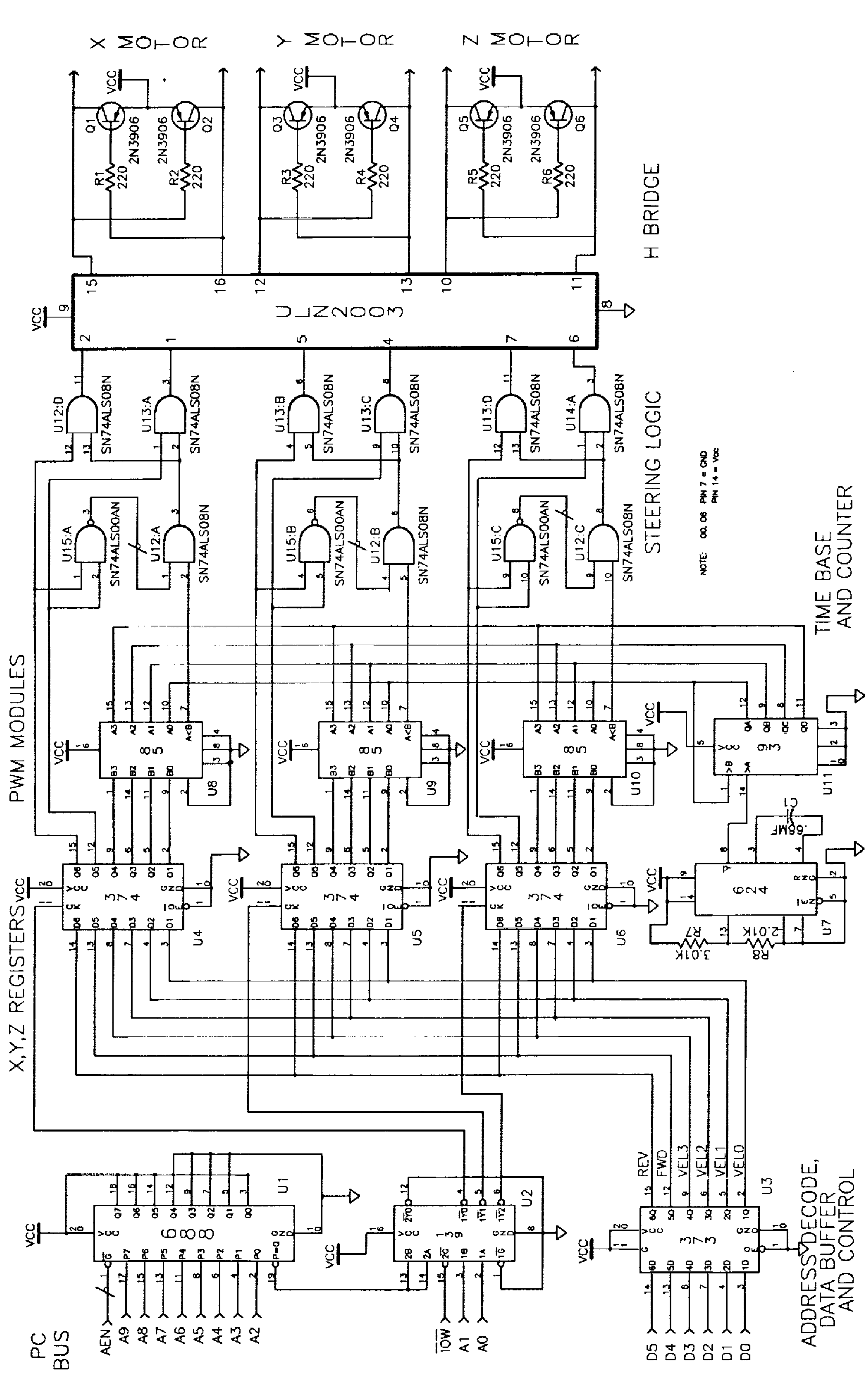
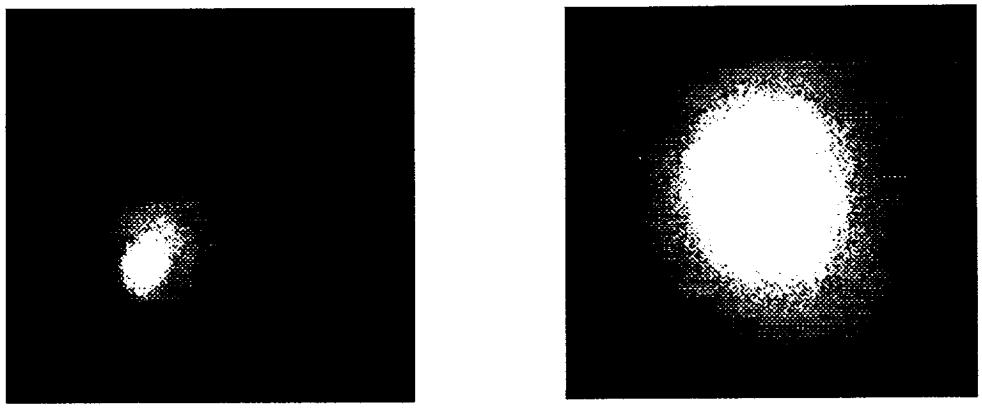
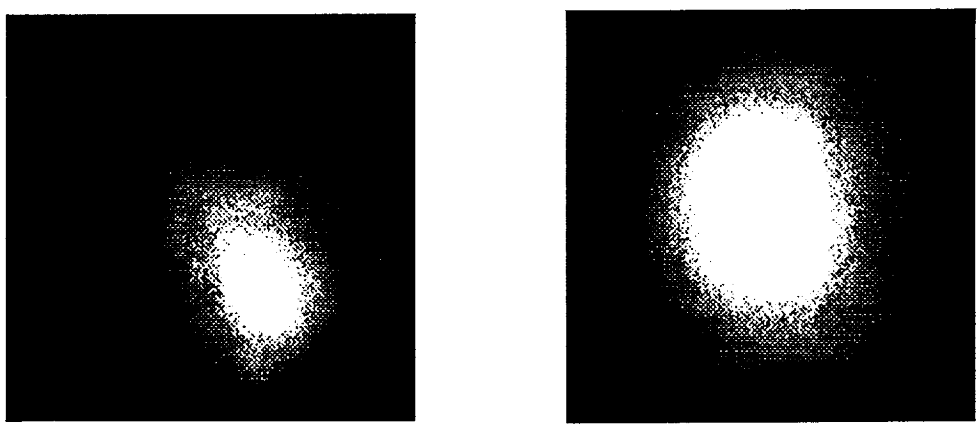

X MOTOR
Y MOTOR
Z MOTOR
H BRIDGE
ULN2003
STEERING LOGIC
NOTE: 00, 08 PIN 7 = GND PIN 14 = Vcc
PWM MODULES
TIME BASE AND COUNTER
X,Y,Z REGISTERS
PC BUS
ADDRESS DECODE, DATA BUFFER AND CONTROL
AEN
A9
A8
A7
A6
A5
A4
A3
A2
IOW
A1
A0
D5
D4
D3
D2
D1
D0
REV
FWD
VEL3
VEL2
VEL1
VEL0
VCC
2N3906
SN74ALS08N
SN74ALS00AN
220
3.01K
2.01K
.68MF

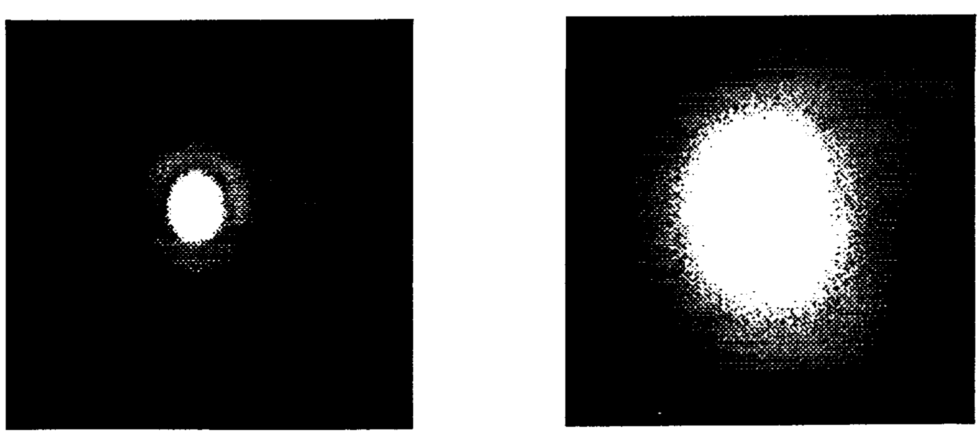
Figure 4.



Example 1



Example 2



Example 3

Figure 5.—Three examples of misaligned initial states on the left with the associated aligned final states on the right.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	December 1992	Technical Memorandum

4. TITLE AND SUBTITLE
A Fuzzy Logic Based Controller for the Automated Alignment of a Laser-Beam-Smoothing Spatial Filter

5. FUNDING NUMBERS
WU-590-21-11

6. AUTHOR(S)
M.J. Krasowski and D.E. Dickens

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135-3191

8. PERFORMING ORGANIZATION REPORT NUMBER
E-7526

9. SPONSORING/MONITORING AGENCY NAMES(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Washington, D.C. 20546-0001

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
NASA TM-105994

11. SUPPLEMENTARY NOTES
M.J. Krasowski, NASA Lewis Research Center, Cleveland, Ohio and D.E. Dickens, Electrical Engineering Department, University of Dayton, Dayton, Ohio 45469. Responsible person, M.J. Krasowski, (216) 433-3729.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified - Unlimited
Subject Categories 31, 35, and 36

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)
A fuzzy logic based controller for a laser-beam-smoothing spatial filter is described. It is demonstrated that a human operator's alignment actions can easily be described by a system of fuzzy rules of inference. The final configuration uses inexpensive, off-the-shelf hardware and allows for a compact, readily implemented embedded control system.

14. SUBJECT TERMS
Fuzzy logic; Fuzzy controller; Computer control; Rule based control; Laser, Spatial filter; Automated alignment; Machine vision

15. NUMBER OF PAGES
16

16. PRICE CODE
A03

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	